**SVEUČILIŠTE U ZAGREBU**

**PRIRODOSLOVNO–MATEMATIČKI FAKULTET**

**MATEMATIČKI ODSJEK**

Ines Frančišković

# BITCOIN

Diplomski rad

Voditelj rada:
Doc. dr. sc. Matija Kazalicki

Zagreb, studeni, 2015

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

# Sadržaj

# Uvod

Bitcoin je prvi primjer kriptovalute. Kriptovaluta je digitalni ekvivalent novca koja funkcionira na temelju kriptografskih algoritama. Jedinice valute zvane bitcoin koriste se za pohranu i prijenos vrijednosti među korisnicima u bitcoin mreži. Korisnici međusobno komuniciraju koristeći bitcoin protokol primarno putem Interneta, iako je moguć prijenos i drugim mrežama. Bitcoin algoritam je napisan kao open source program tako da svatko može vidjeti kod. Bitcoin se može pokretati na raznim računalnim uređajima.

Za razliku od tradicionalnih valuta koje su imale pokriće u prirodnim dobrima, Bitcoin je u potpunosti virtualan. Ne postoji fizički niti digitalni novac kao takav. Novac je utkan u transakcije koje prebacuju vrijednost od pošiljatelja do primatelja. Korisnici mogu prebaciti bitcoine putem mreže te sa njima obavljati gotovo sve što se može obavljati sa tradicionalnim valutama. To uključuje kupovinu i prodaju dobara, slanje novca ljudima ili organizacijama te ostvarivanje kreditne moći. Bitcoini se mogu kupovati, prodavati i zamijeniti za ostale valute (u posebnim mjenjačnicama). U nekom smislu, bitcoin je savršen oblik novca na Internetu jer je brz, siguran i jednostavan za korištenje po cijelom svijetu. Korisnici bitcoina posjeduju ključeve koji im omogućuju da dokažu vlasništvo transakcija u Bitcoin mreži, otključavajući vrijednost koju kasnije mogu potrošiti ili je proslijediti novom primatelju. Ti ključevi su najčešće pohranjeni u digitalnim novčanicima na korisnikovom računalu. Posjedovanje ključa koji otključava transakciju je jedini preduvjet za trošenje bitcoina čime se potpuna kontrola stavlja u ruke svakog korisnika.

Bitcoini se stvaraju kroz proces koji se naziva rudarenje (eng. mining). Rudarenje uključuje traženje rješenja matematičkog problema za vrijeme obrade bitcoin trasakcije. Bilo koji sudionik bitcoin mreže može izvršavati funkciju rudara koristeći računalnu snagu svog računala. Ovi pojedinci ili tvrtke uključuju se u aktivnosti održavanja Bitcoin mreže u zamjenu za transakcijske pristojbe i novonastale bitcoine.

Korisnici mogu slati i primati bitcoine elektroničkim putem bez naknade ili za neznatnu naknadu pomoću računalnih programa tzv. softverskih novčanika (eng. software wallet) koji se mogu nalaziti na osobnom računalu, mobilnom uređaju ili na internetu kao web

aplikacija. Prosječno svakih deset minuta netko je u mogućnosti potvrditi transakcije u proteklih deset minuta i zbog toga je nagrađen novim bitcoinima. Upravo zbog rudarenja bitcoin je decentralizirana valuta te kao takva nije pod kontrolom niti jedne središnje banke.

Bitcoin protokol uključuje ugrađene algoritme koji reguliraju funkciju rudarenja. Složenost problema kojeg rudari moraju riješiti da bi uspješno potvrdili blok transakcija je dinamički prilagođena tako da netko uspije svakih deset minuta neovisno o tome koliko je rudara uključeno u proces. Protokol usporava brzinu stvaranja novih bitcoina za pola svake četiri godine i ograničava ukupan broj bitcoina koji će biti stvoreni na dvadeset jedan milijun. Taj broj biti će dostignut otprilike 2140. godine. Zbog ove ograničene rate izdavanja novih bitcoina, bitcoin ne može doživjeti inflaciju. S obzirom na to da bitcoin ne može pobjeći iz bitcoin mreže, i kako su sve transakcije transparentne, puno je veća vjerojatnost da će bitcoin biti podložan deflaciji. Trenutna vrijednost bitcoina iznosi oko 300$ i podložna je promjenama vrijednosti na dnevnoj bazi, a kroz vrijeme vrijednost bi mu trebala biti stabilnija.

# Poglavlje 1

# Povijest Bitcoina

Bitcoin je predstavljen 2008. godine sa člankom pod imenom "Bitcoin: A Peer-to-Peer Electronic Cash System" [2]. Napisan je pod aliasom Satoshi Nakamoto. Namjera je bila ostvariti potpunu kontrolu nad financijama stvaranjem stabilne, sigurne, svjetski prihvatljive i demokratske valute. Nakamoto je kombinirao nekoliko ranijih izuma kao što su b-money i HashCash kako bi kreirao u potpunosti decentralizirani elektronički platni sustav. Glavna inovacija bila je korištenje distribuiranog sustava izračunavanja ("proof-of-work" algoritam) kako bi se potvrdila transakcija. Vrijeme potrebno za potvrđivanje transakcije iznosi otprilike deset minuta. Ovaj način dopušta decentraliziranoj mreži da stigne na konsenzus o stanju transakcije čime se spriječilo dvostruko trošenje koje je u prošlosti bilo velika mana digitalnih valuta.

Bitcoin mreža je započela sa radom 2009. Bazirana je na implementacijskoj referenci objavljenoj od Nakamota. U samim početcima Bitcoin mreža je bila sklona pogreškama. U 2009. godini pronađena je pogreška u ranom Bitcoin klijentu što je dopustilo stvaranje velikog broja bitcoina. U ožujku 2013. tehnički kvar je izazvao podjelu Bitcoin mreže na dva neovisna dijela. Šest sati dvije različite Bitcoin mreže su radile paralelno, svaka sa svojom verzijom povijesti transakcija. Programeri Bitcoin sustava pozvali su na privremenu obustavu prometa, što je izazvalo oštar pad cijene. Normalni način rada je obnovljen kroz nekoliko sati.

Veće web stranice počele su prihvaćati bitcoin otprilike 2013. WordPress je započeo u studenom 2012. nakon čega slijede OKCupid u travnju 2013., Atomic Mall u studenom 2013., TigerDirect i Overstock.com u siječnju 2014. te Expedia u lipnju 2014. Određene neprofitne ili interesne skupine kao što su Electronic Frontier Foundation dopuštaju bitcoin donacije. Vlasnik trgovine Overstock.com je u ožujku 2014. izjavio da ta trgovina ima dnevni promet od dvadeset do trideset tisuća dolara od kupaca koji plaćaju bitcoinima.

Prve veće policijske i pravosudne akcije protiv onih koji zloupotrebljavaju Bitcoin sustav kreću u svibnju 2013. u SAD-u. Imovina koja pripada burzi Mt.Gox zaplijenjena je od strane američkog Ministarstva domovinske sigurnosti, a ilegalna trgovina Silk Road je zatvorena od strane FBI-a.

U listopadu 2013. kineski internetski gigant Baidu je uveo plaćanje bitcoinima. Tijekom studenog 2013. Bitcoin burza "BTC China" osnovana u Kini pretekla je japansku burzu Mt. Gox i europsku burzu Bitstamp te postala najveća Bitcoin burza po obujmu trgovine. Dana 19. studenoga 2013. vrijednost bitcoina na burzi Mt. Gox je skočila na vrhunac od 900$ nakon što je na ročištu odbora Senata SAD-a bilo rečeno da su virtualne valute legitimna financijska usluga. Na isti dan jedan Bitcoin je vrijedio više od 6780 RMB(1100 $) u Kini. Dana 5. prosinca 2013. Narodna banka Kine je zabranila kineskim financijskim institucijama korištenje bitcoin valutnog sustava. Nakon objave vrijednost bitcoina je pala i Baidu više ne prihvaća bitcoine za određene usluge.

Najveća do sada zabilježena transakcija iznosila je sto pedeset milijuna dolara koji su prebačeni momentalno i obrađeni bez ikakvih manipulativnih troškova.
Prvi Bitcoin ATM je instaliran u listopadu 2013. u Vancouveru u Kanadi.

Trenutni broj korisnika Bitcoina nemoguće je točno utvrditi. Razlog tome je taj što jedan korisnik može imati više adresa. Na osnovu broja korisnika aplikacija za bitcoin transakcije, tzv. softverskih novčanika pretpostavlja se da taj broj danas iznosi oko osam milijuna.

Distribuirane jedinice koje pružaju sigurnost i fleksibilnost u bitcoin mreži su se eksponencijalno povećale i sada dostižu kombiniranu snagu obrade najjačih svjetskih super računala. Vrijednost bitcoina na tržištu iznosi između predviđenih pet i deset milijardi američkih dolara, ovisno o dnevnim oscilacijama u zamjenama bitcoin-dolar.

Satoshi Nakamoto se povukao iz javnosti u travnju 2011. ostavivši odgovornost razvijanja koda i mreže grupi nadobudnih volontera. Identitet osobe ili grupe koja stoji iza Bitcoina još uvijek je nepoznat. Unatoč tome, niti Satoshi Nakamoto niti itko drugi ne zadržava kontrolu nad Bitcoin sustavom koji djeluje po potpuno transparentnim matematičkim principima. Sama ideja je revolucionarna i već je započela novu znanost u poljima distributivnih računalnih znanosti, ekonomije i ekonometrije.

# Poglavlje 2

# Bitcoin protokol

U ovom poglavlju opisati ću glavne ideje na kojima se bazira Bitcoin protokol [3]. Zbog lakšeg razumijevanja Bitcoin ću razviti u fazama. Prvo ću opisati vrlo jednostavnu digitalnu valutu koju ću nazvati Infocoin. Infocoin će na početku imati mnogo nedostataka, ali u svakom koraku dodati ću joj neku funkcionalnost. Nakon određenog broja koraka Infocoin će biti gotovo jednak Bitcoin protokolu.

## 2.1 Kako možemo konstruirati digitalnu valutu?

Zamislimmo da neka osoba posjeduje digitalni novac kojeg želi potrošiti. Nazovimo tu osobu Ana. Pretpostavimo da je Anin digitalni novac predstavljen stringom bitova. Prva dva problema na koja nailazimo su: Kako se može spriječiti da ona taj isti string koristi više puta tako stvarajući beskonačnu zalihu novca? Kako spriječiti nekoga da krivotvori taj string bitova i koristi ga kako bi krao od Ane? To su samo dva od mnogih problema koji se moraju savladati kako bi se podaci mogli koristiti kao novac.

U prvoj verziji Infocoina pokušati ću pronaći način da Ana koristi string bitova kao vrlo primitivni oblik novca koji bi joj omogućio barem neku zaštitu protiv krivotvorenja. Zamislimo da Ana želi nekoj osobi, nazovimo ju Petra, dati infocoin. Kako bi to učinila Ana piše poruku "Ja Ana, dajem Petri jedan infocoin". Ana zatim stvara digitalni potpis koristeći tajni kriptografski ključ te svima objavljuje potpisani string bitova. Više o kriptografskim ključevima u dodatku A.1.
Ovo nije neki impresivni prototip digitalnog novca ali ipak ima neke vrline. Svi ostali korisnici, pa tako i Petra, mogu koristiti Anin javni ključ kako bi provjerili da je zaista Ana stvorila poruku "Ja Ana, dajem Petri jedan infocoin". Nitko drugi nije mogao stvoriti upravo taj string pa zbog toga Ana ne može reći da nije mislila dati Petri infocoin. Činjenica da nitko drugi nije mogao stvoriti takvu poruku daje Ani neku ograničenu zaštitu protiv kri-

votvorenja. Krivotvorenje je nažalost ipak moguće u slučaju da nakon što Ana objavi svoju poruku mreži netko od korisnika kopira tu poruku. Ova dva svojstva: utvrđivanje namjere od strane Ane i ograničena zaštita od krivotvorenja su najznačajnije značajke ove verzije protokola. Važno je još napomenuti kako je digitalni novac u ovom primjeru zapravo string bitova koji predstavljaju digitalno potpisanu poruku: "Ja Ana, dajem Petri jedan infocoin".

Oblik novca te princip rada u sljedećim verzijama protokolima biti će isti. Razlika će biti u tome što će poruke biti sve složenije.

## Korištenje serijskih brojeva kako bi se novčanice mogle jedinstveno identificirati

Problem u prvoj verziji Infocoina bio je u tome što je Ana mogla Petri više puta poslati jednu te istu poruku. Zamislimo da je Petra primila deset kopija poruke "Ja Ana, dajem Petri jedan infocoin". Znači li to da je Ana poslala Petri deset različitih infocoina? Možda je samo pokušala navesti Petru da misli da joj je dala deset različitih infocoina iako poruka dokazuje da joj namjerava dati samo jedan infocoin.

Cilj je pronaći način kojim bi jedinstveno identificirali infocoine. Pretpostavimo da svaki infocoin ima svoj serijski broj. Tada bi Ana poslala poruku "Ja Ana, dajem Petri jedan infocoin, sa serijskim brojem 8740348". Nakon toga bi mogla poslati poruku "Ja Ana, dajem Petri jedan infocoin, sa serijskim brojem 8770431". Tada bi Petra ali i svi ostali korisnici znali da je prebačen drugi infocoin. Kako bi ovaj način rada ispravno funkcionirao potreban je pouzdan izvor serijskih brojeva za infocoine. Jedna od mogucnosti je uvođenje banke. Uloga banke bila bi dodjeljivanje infocoinima serijske brojeve, evidentiranje kod koga se koji infocoini nalaze te potvrđivanje valjanih transakcija.

Pretpostavimo da Ana ode u banku i kaže da želi podići jedan infocoin. Banka će tada smanjiti stanje na njenom računu za jedan infocoin, te infocoinu kojeg želi podići dodijeliti novi, nikada prije korišteni serijski broj, npr. 1234567. Ukoliko bi sada Ana željela Petri dati jedan infocoin poslala bi poruku "Ja Ana, dajem Petri jedan infocoin, sa serijskim brojem 1234567". Petra bi prije nego što prihvati infocoin otišla u banku da provjeri da li infocoin sa serijskim brojem 1234567 zaista pripada Ani te da Ana možda nije ranije potrošila taj infocoin. Ukoliko je sve u redu Petra bi rekla banci da želi prihvatiti infocoin. Nakon toga banka bi ažurirala podatke kako bi infocoin sa serijeskim brojem 1234567 bio u Petrinom vlasništvu.

## Eliminacija središnjeg autoriteta - banke

Iako posljednje rješenje djeluje obećavajuće ono se može još poboljšati eliminacijom banke. Ideja je da svi korisnici zajedno čine banku.

Pretpostaviti ćemo da svi korisnici Infocoina posjeduju zapis svih transakcija. Taj zapis nazvati ćemo Blockchain. Pretpostavimo da Ana želi dati Petri jedan infocoin. To će učiniti tako što će joj poslati poruku "Ja Ana, dajem Petri jedan infocoin, sa serijskim brojem 1234567". Petra koristi svoju kopiju Blockchaina kako bi provjerila da je Ana zaista vlasnica infocoina sa serijskim brojem 1234567. Ukoliko se to ispostavi točnim Petra svim korisnicima objavljuje Aninu poruku te svoje prihvaćanje transakcije. Nakon toga svi korisnici ažuriraju svoj Blockchain.

Problem u ovoj verziji protokola je dvostruko trošenje. Pretpostavimo da Ana pošalje Petri poruku "Ja Ana, dajem Petri jedan infocoin, sa serijskim brojem 1234567" te nedugo nakon toga Marku poruku "Ja Ana, dajem Marku jedan infocoin, sa serijskim brojem 1234567". I Petra i Marko će koristiti svoju kopiju Blockchaina kako bi provjerili da li Ana zaista posjeduje infocoin sa serijskim brojem 1234567. Ukoliko oni tu provjeru obavljaju otprilike istovremeno oboje će prihvatiti transakciju te nakon toga mreži objaviti svoj prihvat transakcije. Na koji način će u tom slučaju ostali korisnici ažurirati svoj Blochchain? Čak i ukoliko se svi korisnici nekako uspiju dogovoriti o izgledu BlockChaina ili Marko ili Petra biti će prevareni.

Kako uspješno rješiti problem dvostrukog trošenja? Rješenje leži u činjenici da kada Petra dobije infocoin ne pokušava sama provjeravati valjanost transakcije već da moguću transakciju objavi svim Infocoin korisnicima kako bi joj pomogli odlučiti da li je transacija valjana. Ukoliko dovoljan broj korisnika odluči da je transakcija valjana, Petra ju prihvaća. Nakon toga svi korisnici moraju ažurirati svoje kopije Blockchaina.

Ova verzije protokola sprječava dvostruko trošenje jer ukoliko Ana pokuša poslati isti infocoin Petri i Marku ostali kosrisnici će to primjetiti pa transakcija neće biti potvrdena.

Ova verzija protokola ima dosta nejasnih dijelova kao npr. Odakle dolaze jedinstveni serijski brojevi? Što znači dovoljan broj korisnika kod provjeravanja valjanosti transakcije? On sigurno ne predstavlja ukupan broj svih korisnika na mreži jer njega ne možemo točno utvrditi. Iz istog razloga on ne može biti niti neki postotak ukupnog broja korisnika na mreži.

## Proof-of-work

Pretpostavimo da Ana želi dvostruko trošiti u posljednjoj opisanoj verziji protokola. To bi mogla učiniti jedino ukoliko preuzme kontrolu nad cijelom infocoin mrežom. Pretpostavimo da Ana koristi automatizirani sustav kako bi postavila veliki broj odvojenih identiteta u infocoin mreži, npr. milijardu. Kao i u prethodnom primjeru, pokušati će poslati isti infocoin Petri i Marku. Aninini lažni identiteti na mreži objaviti će da su obje transakcije valjane i tako navesti i Petru i Marka na prihvaćanje transakcije što znači da se desila prevara.

Postoji način kojim se može izbjeći ovaj problem i on se naziva proof-of work. Proof-of-work je nastao kombinacijom dviju ideja: Prva je ta da za potvrđivanje transakcije bude potrebna računalna snaga. Druga je ta da se korisnik nagradi za sudjelovanje u prosecu potvrđivanja određene transakcije. Nagrada služi kao motivacija korisnicima da se uključe u proces potvrđivanja transakcija iako je za to potrebna računalna snaga. Prednost toga da potvrđivanje transakcije iziskuje računalnu snagu očituje se u činjenici da postaje gotovo nemoguće da na potvrdu transakcije utječe veliki broj računalnih identiteta kojima upravlja jedna osoba. Da bi se prevara ipak uspjela desiti potrebni su vrlo veliki računalni resursi.

Pretpostavimo da Ana objavi mreži poruku: ”Ja, Ana, dajem Petri jedan infocoin sa serijskim brojem 1234567.” Kada ostali korisnici prime tu poruku, svaki ju dodaje u svoj red neriješenih transakcija. To su transakcije za koje je korisnik čuo, ali nisu još potvrdene od strane mreže. Npr. pretpostavimo da postoji korisnik David koji u redu neriješenih transakcija ima sljedeće transakcije :
”Ja, Tom, dajem Sanji jedan infocoin sa serijeskim brojem 1201174.”
”Ja, Saša, dajem Klari jedan infocoin sa serijeskim brojem 1295618.”
”Ja, Ana, dajem Petri jedan infocoin sa serijeskim brojem 12334567.”
David će prvo provjeriti svoju kopiju Blockchaina te vidjeti da je svaka od navedenih transakcija valjana. On bi htio pomoći ostalim korisnicima tako da im objavi vijest o valjanosti transakcija. No prije nego što to učini kao dio procesa potvrdivanja transakcije mora riješiti složeni računali problem, tzv. proof-of-work. Bez rješenja tog problema ostatak mreže neće prihvatiti njegovu potvrdu transakcije. Kakvu zagonetku Davit treba riješiti?

Neka je h fiksna hash funkcija poznata svim korisnicima - ugrađena u protokol. Bitcoin koristi SHA-256 hash funkciju, ali odgovarala bi i svaka druga kriptografski sigurna hash funkcija. Više o hash funkcijama u dodatku B. Davidov red neriješenih transakcija označiti ćemo sa l. Pretpostavimo da David doda broj x nizu l te primjeni hash funkciju. Npr. ukoliko koristimo l=”Hello world” (očito nije transakcija već se tu nalazi zbog primjera) i x=0 izlazna vrijednost će biti

```
h("Hello, world!0") =

    1312af178c253f84028d480a6adc1e25e81caa44c749ec81976192e2ec934c64
```

Problem kojeg David mora riješiti - proof-of-work sastoji se od traženja broja x za kojeg vrijedi da kada ga dodamo stringu l pa primjenimo hash funkciju kao rezultat dobijemo string koji započinje sa nizom nula. Problem može biti jednostavniji ili kompliciraniji, ovisno o broju nula s kojima rješenje mora započinjati.Jedostavnije zagonetke zahtijevaju da string započinje sa tri ili četiri nule dok one složenije zahtijevaju npr. petnaest. U ona slučaja, prethodni pokušaj pronalaska rješenja za x=0 je neuspješan, kao i onaj za x=1.

```
h("Hello, world!1") =

    e9afc424b79e4f6ab42d99c81156d3a17228d6e1eef4139be78e948a9332a7d8
```

Redom pokušavamo za vrijednosti 2, 3,... Končno, za x=4250 pronalazimo valjano rješenje.

```
h("Hello, world!4250") =

    0000c3af42fc31103f1fdc0151fa747ff87349a4714df7cc52ea464e12dcd4e9
```

Za x=4250 primjenom hash funkcije dobijemo string koji započinje s četiri nule.To je dovoljan broj nula za jednostavniji proof-of-work problem , ali ne i za one složenije.

Ono što problem čini teškim za riješiti je činjenica da se izlazni rezultat kriptografske hash funkcije ponaša kao slučajni broj: ukoliko samo malo promijenimo argument hash funkcije, izlazna vrijednosti će se u potpunosti promijeniti na način koji je nomoguće predvidjeti. Ukoliko želimo da izlazni rezultat započinje s deset nula, David će u prosjeku uvrstiti $16^{10} \approx 10^{12}$ različitih vrijednosti za x prije nego uspije pronaći odgovarajuće rješenje. To je poprilično izazovan zadatak, koji zahtijeva veliku računalnu snagu.

Bitcoin protokol služi se malo drugačijim problemom kojeg se mora riješiti ukoliko se transakcija želi potvrditi. Umjesto zahtjeva da izlazni rezultat hash funkcije započinje s odredenim brojem nula, Bitcoin proof-of-work zagonetka zahtijeva da hash zaglavlja bloka bude manji ili jednak od broja koji se naziva cilj. Cilj se automatski prilagođava kako bi se osiguralo da vrijeme potrbno za prihvaćanje bloka bude otprilike deset minuta.

Pretpostavimo da David uspije pronaći broj x koji zadovoljava uvjete problema. David će tada objaviti blok transakcija koje on odobrava, zajedno sa vrijednosti x-a. Ostali sudionici u mreži mogu lako provjeriti da je x valjano rješenje proof-of-work problema. Nakon toga oni ažuriraju svoj Blockchain tako da uključuje novi blok transakcija koji je David

potvrdio.

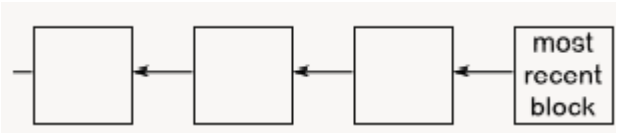
Kako bi proof-of-work ideja imala ikakve šanse za uspjehom, korisnicima je potreban poticaj za sudjelovanje u procesu potvrđivanja transakcije. Bez tog poticaja oni ne bi imali razloga da troše računalnu snagu pokušavajući pronaći rješenje i cijeli sustav ne bi radio.

Ovaj problem se rješava tako da se svaki korisnik koji uspije potvrditi novi blok nagradi sa infocoin-ima.

U Bitcoin protokolu se proces kojim se potvrđuju transakcije naziva rudarenje. Za svaki potvrđeni blok transakcija uspješni rudar prima nagradu u bitcoinima. U početku je nagrada iznosila pedeset bitcoina. Za otprilike svakih 210000 potvrdenih blokova nagrada se prepolavlja. To se do sada desilo samo jednom pa trenutna nagrada za rudarenje iznosi dvadeset pet bitcoina. Ovo prepolavljanje nagrade događati će se otprilike svake četiri godine do 2140. U tom trenutku nagrada za rudarenje će pasti ispod $10^{-8}$ bitcoina po bloku. $10^{-8}$ je zapravo najmanja jedinica Bitcoina koja se naziva Satoshi. Zbog toga će 2140.godine ukupna količina bitcoina prestati rasti. Unatoč tome rudari neće prestati dobivati nagradu za potvrđivanje transakcije. Bitcoin omogućuje da određena količina novca u transakciji služi kao naknada rudaru koji je pomogne potvrditi. U počecima Bitcoina transakcijska naknada je uglavnom bila postavljena na nulu, ali kako je popularnost Bitcoina rasla postupno se povećavala i transakcijska naknada te danas ona služi rudarima kao dodatna motivacija.

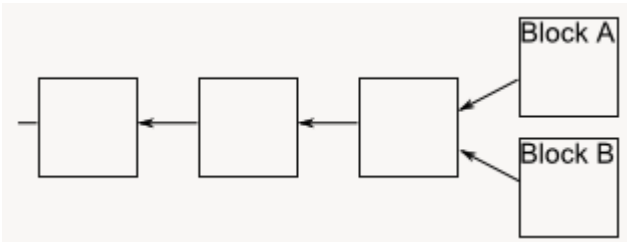
Proof-of-work se jednostavno može shvatiti kao natjecanje za odobravanje transakcija. Svaki ulaz u natjecanje košta odredenu količinu računalne snage. Rudareva šansa za pobjedom u natjecanju je proporcionalna količini računalne snage uložene u izračunavanje. Zbog činjenice da su u natjecanje uvedeni veliki računalni resursi, nepošten rudar ima relativno male šanse da bi omeo proces potvrdivanja transakcije, osim ako ne posjeduje veliku količinu računalnih resursa.

Infocoin mreža trebala bi se jednoglasno složiti oko poretka u kojem su se transakcije desile. Ukoliko jasan poredak ne postoji, u nekom trenutku može se desiti da nije u potpunosti jasno tko posjeduje koje infocoine. Zbog toga se zahtjeva da novi blok uvijek sadrži pokazivač na prethodni potvrdeni blok. Pokazivač će zapravo biti hash prethodnog bloka. Blockchain će tada zapravo biti linearan lanac blokova transakcija, u kojem svaki blok sadrži pokazivač na prethodni blok.




Slika 2.1: Lanac blokova

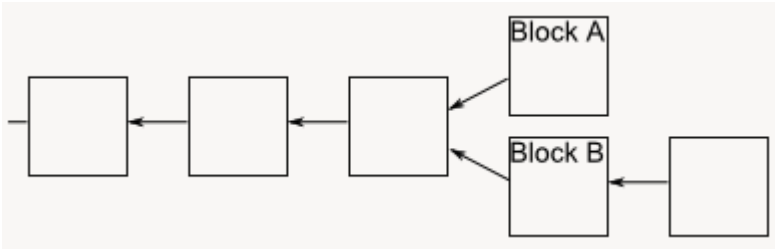
Ponekad se u Blockchainu dogodi račvanje. To se može dogoditi ukoliko dva rudara gotovo istovremeno potvrde blok transakcija i tu vijest simulatano objave mreži. Tada će neki korisnici ažurirati svoj Blockchain na jedan način, a neki na drugi.




Slika 2.2: Račvanje

Ovo uzrokuje problem jer više nije točno jasno u kojem su se poretku desile transakcije. Na sreću postoji jednostavan način da se uklone račvanja. Pravilo je sljedeće: Ukoliko dođe do račvanja korisnici će pratiti oba dijela lanca, ali će u svakom trenutku rudari raditi na proširivanju dijela lanca koji je u njihovoj kopiji Blokchaina dulji.

Pretpostavimo npr. da postoji račvanje u kojem neki rudari prvo prime blok A, a neki blok B. Rudari koji prvo prime blok A nastaviti će rudariti po tom dijelu, a oni ostali po drugom dijelu. Pretpostavimo da će rudari koji rade na B dijelu sljedeći uspješno rudariti blok:




Slika 2.3: Proširivanje B dijela

Nakon što prime vijest da se ovo desilo, rudari koji rade na A dijelu primjetiti će da je B dio dulji i prebaciti će se raditi na tom dijelu. Nedugo zatim prekinuti će se rad na A dijelu, svi rudari će raditi na istom linearnom lancu i blok A će biti ignoriran. Naravno, sve transakcije koje se nalaze u redu čekanja u A će se nalaziti u redovima čekanja rudara koji rade na B i s vremenom će biti potvrđene.

Također se može desiti da rudari koji rade na A dijelu budu prvi koji će proširiti svoj dio. U tom slučaju prekinuti će se rad na B dijelu te ćemo ponovno imati jedan linearan lanac.

Neovisno o ishodu ovaj način osigura da su blokovi u lancu složeni prema vremenu potvrđivanja. U Bitcoion protokolu transakcija se ne smatra potvrđenom dok nisu zadovoljena sljedeća dva uvjeta: (1) dio je bloka koji se nalazi u najduljem dijelu; (2) nakon bloka u kojem se ona nalazi slijedi barem još pet blokova. U tom slučaju kažemo da transakcija ima barem ”šest potvrda”. Ta činjenica daje mreži vremena da se dogovori oko rasporeda blokova u Blockchain. Ovu strategiju koristiti ćemo u Infocoinu.

Sada kada je razjašnjeno slaganje blokova u Blockchain vratiti ćemo se na problem dvostrukog trošenja. Pretpostavimo da Ana želi poslati isti infocoin Petri i Marku. Jedan mogući pristup bio bi da Ana pokuša potvrditi blok kloji uključuje obe transakcije. Ako pretpostavimo da ona posjeduje jedan posto računalne snage svako toliko će uspjeti potvrditi blok tako što će rješiti proof-of-work. Međutim, na Aninu nesreću dvostruko trošenje biti će odmah uočeno od strane ostalih korisnika Infocoin mreže i odbijeno unatoč rješenom proof-of-work problemu. Zbog toga ovo nije nešto oko čega bi se trebali brinuti.

Mnogo veći problem nastaje ukoliko Ana objavi dvije zasebne transakcije u kojima ona šalje isti infocoin Petri i Marku. Mogla bi to učiniti tako da jednu transakciju objavi jednoj skupini rudara, a drugu nekoj drugoj skupini rudara nadajući se da će se obje transakcije uspjeti potvrditi. Na sreću u ovom slučaju će mreža, kao što smo ranije vidjeli potvrditi

jednu od ovih transakcija, ali ne i obje. Ukoliko npr. Marko vidi da njegova transakcija nije potvrdena, odbiti će Aninu ponudu. Vidimo da i ovaj pokušaj prevare nije nešto oko čega bi se trebali brinuti.

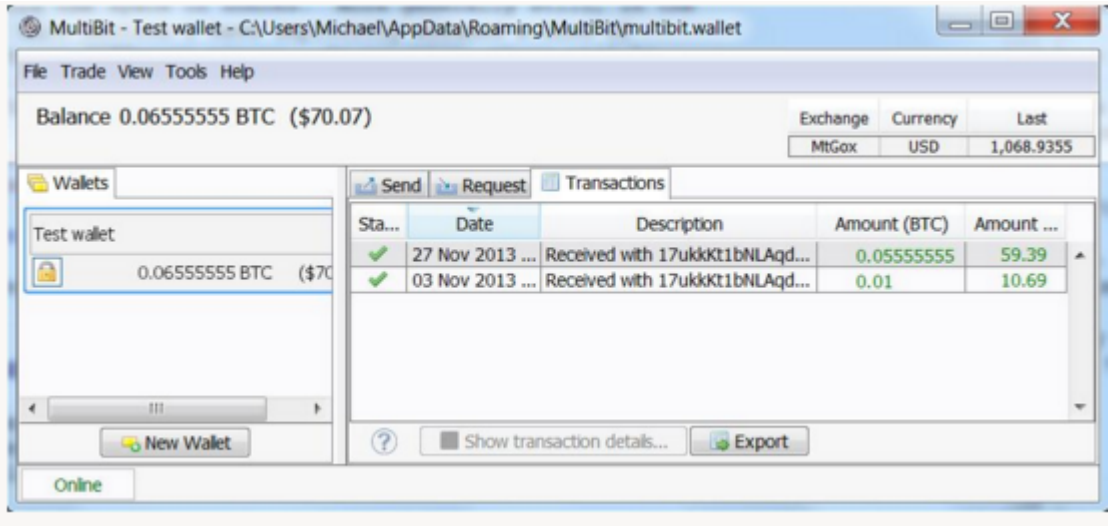
Važna varijanta dvostrukog trošenja dogoditi će se ukoliko je Ana=Petra, tj. ukoliko Ana pokuša isti infocoin poslati Marku i sebi samoj. Ovo se čini kao nešto što se lako može otkriti i s čime se lako može nositi, no međutim nije teško na mreži postaviti višestruke identitete povezane sa istom osobom ili organizacijom. U ovom slučaju Anina strategija biti će da čeka dok Marko ne prihvati infocoin, što će se desiti nakon što se transakcija potvrdi 6 puta na najduljem lancu. Ona će tada pokušati račvati lanac prije transakcije s Markom tako što će dodati blok koji uključuje transakciju kojom sama sebi plaća infocoin.

Na Aninu žalost, sada je vrlo teško za nju da sustigne dulji dio lanca. Ostali rudari joj neće pomoći jer će oni raditi na duljem dijelu. Jedino u slučaju da Ana uspije riješiti proof-of-work barem jednako brzo kao svi ostali korisnici zajedno što bi značilo da mora kontrolirati otprilke pedeset posto računalne snage infocoin mreže. Možemo zamisliti scenarij u kojem Ana posjeduje jedan posto računalne snage ali joj se posreći da nade šest blokova jedan za drugim prije nego ostatk mreže pronade neki dodatni blok. U tom slučaju, Ana može dobiti kontrolu nad Blockchainom. Međutim vjerojatnost da se ovo dogodi je $1/10^6 = 10^{-12}$.

## 2.2 Primjer transakcije u Bitcoin mreži

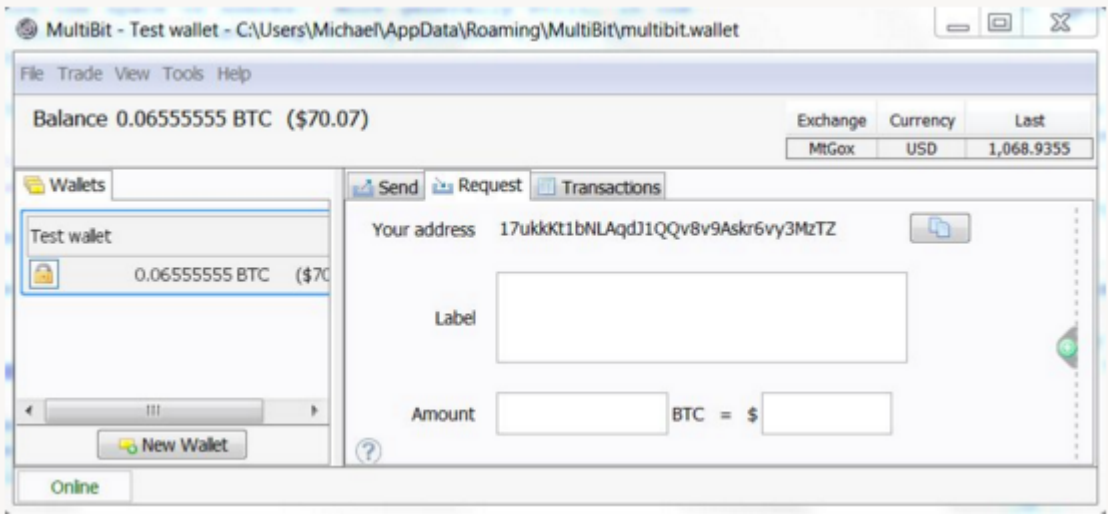
Sada ću opisati primjer transakcije u stvarnoj Bitcoin mreži.

Kako bi koristili Bitcoin moramo na računalo instalirati aplikaciju koja se naziva softverski novčanik. Postoji više različitih softverskih novčanika. Na slici 2.4 nalazi se primjer snimke zaslona novčanika koji se naziva Multbit. Na lijevoj strani može se vidjeti stanje računa koje iznosi 0.06555555 bitcoina, a na desnoj posljednje dvije transakcije.




Slika 2.4: primjer sučelja novčanika koji se naziva Multibit

Pretpostavimo da neki trgovac želi postaviti internet trgovinu u kojoj će se moći plaćati bitcoinima. Da bi to bilo moguće prvo mora pomoću aplikacije generirati Bitcoin adresu. Kako bi se dobila Bitcoin adresa prvo se stvara privatni i javni ključ, a zatim primjeni hash funkcija na vrijednost javnog ključa.




Slika 2.5: Bitcoin adresa

Ukoliko netko želi obaviti kupnju u toj trgovini potrebna mu je Bitcoin adresa trgovca. Trgovac može svoju Bitcoin adresu poslati kupcu mailom ili ju čak javno objaviti na web stranici. To je sigurno jer je adresa samo hash vrijednost javnog ključa.

Nakon što sazna Bitcoin adresu trgovca kupac će generirati transakciju. Na slici 2.5 nalaze se podaci stvarne transakcije kojom se prebacuje 0.31900000 bitcoina. Zbog jednostavnosti prikazano je samo prvih šest znakova hash vrijednosti i dodani su brojevi linija.

```
1.  {"hash":"7c4025...",
2.  "ver":1,
3.  "vin_sz":1,
4.  "vout_sz":1,
5.  "lock_time":0,
6.  "size":224,
7.  "in":[
8.    {"prev_out":
9.      {"hash":"2007ae...",
10.      "n":0},
11.    "scriptSig":"304502... 042b2d..."}],
12. "out":[
13.   {"value":"0.31900000",
14.    "scriptPubKey":"OP_DUP OP_HASH160 a7db6f OP_EQUALVERIFY OP_CHECKSIG"}]}
```

Slika 2.6: Bitcoin transakcija

Prva linija sadrži hash vrijednost ostatka transakcije. Ona se koristi kao identifikator transakcije.

Druga linija nam govori da se promatrana transakcija nalazi u verziji 1 Bitcoin protokola.

Iz treće i četvrte linije vidimo da transakcija ima jednu ulaznu vrijednost i jednu izlaznu.

Peta linija sadrži vrijednost varijable lock_time koja određuje kada će transakcija biti izvršena. Za većinu transakcija u Bitcoin mreži ona je postavljena na nulu, što znači da će transakcije biti odmah izvršene.

Linija šest sadrži veličinu transakcije u bajtovima. važno je primjetiti da to nije vrijednost bitcoina koji se prebacuju.

Linije sedam do jedanaest definiraju ulazne vrijednosti transakcije. Linije osam do deset govore da je ulazna vrijednost ove transakcije uzeta iz izlazne vrijednosti neke druge transakcije sa danom hash vrijednosti $2007ae$. $n = 0$ znači da je to bila prva izlazna vrijednost u danoj transakciji jer transakcije mogu imati više ulaznih i izlaznih vrijednosti. Jedanaesta linija sadrži potpis osobe koja šalje novac, 304502... nakon čega slijedi razmak te odgovarajući javni ključ, $04b2d$. Važno je primjetiti da se nigdje ne definira koliko će se bitcoina iz prethodne transakcije potrošiti u ovoj transakciji. Zapravo će se potrošiti svi bitcoini iz nulte izlazne vrijednosti prethodne transakcije. To se može činiti kao nepovoljno ograničenje, kao npr. da kupimo kruh novčanicom od sto kuna i nismo u mogućnosti dobiti natrag ostatak. Ovaj problem rješava se transakcijama koje imaju više ulaznih i izlaznih vrijednosti.

Linije dvanaest do četrnaest definiraju izlaznu vrijednost transakcije. Trinaesta linija sadrži izlaznu vrijednost koja za ovu transakciju iznosi 0.319 bitcoina. String $a7db6f$ je Bitcoin adresa primatelja.

Odakle dolaze serijski brojevi bitcoina? Ulogu serijskih brojeva bitcoina zapravo imaju hash vrijednosti transakcija. U transakciji na slici 2.5 trgovac prima 0.319 bitcoina, koji su došli iz prve izlazne vrijednosti transakcije čija je hash vrijednost $2007ae$. Ukoliko u Blockchainu potražimo tu transakciju vidjet ćemo da njena izlazna vrijednost dolazi iz neke još ranije transakcije i tako dalje.

Moguće je pratiti lanac transakcija dublje u povijest. Nakon nekog vremena taj proces će završiti. To se može dogoditi iz jednog od dva razloga. Prvi je taj da smo došli

do prve Bitcoin transakcije sadržane u početnom bloku (Genesis blok). Ta transakcija je specifična jer nema ulazne vrijednosti, već samo izlaznu vrijednost koja iznosi pedeset bitcoina. Time je određena početna zaliha novca. Druga mogućnost je ta da dođemo do posebne transakcije koja se naziva coinbase. Svaki blok transakcija osim početnog bloka započinje sa coinbase transakcijom. To je transakcija kojom se nagrađuje uspješni rudar u procesu rudarenja.

# Dodatak A

# Kriptosustavi s javnim ključem

## A.1 Osnovni pojmovi iz kriptografije

Kriptografija je znanstvena disciplina koja se bavi proučavanjem metoda za slanje poruka u takvom obliku da ih samo onaj kome su namijenjene može pročitati. Osnovni zadatak kriptografije je omogućiti dvjema osobama (pošiljalac i primalac) komuniciranje preko nesigurnog komunikacijskog kanala na način da treća osoba (njihov protivnik), koja može nadzirati komunikacijski kanal, ne može razumjeti njihove poruke. Poruku koju pošiljalac želi poslati primaocu zvat ćemo otvoreni tekst. Pošiljalac transformira otvoreni tekst koristeći unaprijed dogovoreni ključ. Taj postupak se naziva šifriranje, a dobiveni rezultat šifrat ili kriptogram. Nakon toga pošiljalac pošalje šifrat preko nekog komunikacijskog kanala. Protivnik prisluškujući može doznati sadržaj šifrata, ali ne može odrediti otvoreni tekst. Za razliku od njega, primalac koji zna ključ kojim je šifrirana poruka može dešifrirati šifrat i odrediti otvoreni tekst. Kriptografski algoritam ili šifra je matematička funkcija koja se koristi za šifriranje i dešifriranje. Općenito, radi se o dvije funkcije, jednoj za šifriranje, a drugoj za dešifriranje. Te funkcije preslikavaju osnovne elemente otvorenog teksta (najčešće su to slova, bitovi, grupe slova ili bitova) u osnovne elemente šifrata, i obratno. Funkcije se biraju iz određene familije funkcija u ovisnosti o ključu. Skup svih mogućih vrijednosti ključeva nazivamo prostor ključeva. Kriptosustav se sastoji od kriptografskog algoritma, te svih mogućih otvorenih tekstova, šifrata i ključeva.

**Definicija A.1.1.** *Kriptosustav je uređena petorka (P, C, K, E, D) za koju vrijedi:*

1. *P je konačan skup svih mogućih osnovnih elementa otvorenog teksta;*

2. *C je konačan skup svih mogućih osnovnih elemenata šifrata;*

3. *K je prostor ključeva, tj. konačan skup svih mogućih ključeva;*

4. *Za svaki $K \in K$ postoji funkcija šifriranja $e_k \in E$ i odgovarajuća funkcija dešifriranja $d_k \in D$. Pritom su $e_k : P \to C$ i $d_k : C \to P$ funkcije sa svojstvom da je $d_k(e_k(x)) = x$*

Kriptosustavi se s obzirom na tajnost ključeva djele na kriptosustave s tajnim ključem i kriptosustave s javnim ključem.Kod kriptosustava s javim ključem, ključ za dešifriranje se može izračunati poznavajući ključ za šifriranje i obratno. Sigurnost ovih kriptosustava leži u tajnosti ključa. Kod kriptosustava s javnim ključem, ključ za dešifriranje se ne može (barem ne u nekom razumnom vremenu) izračunati iz ključa za šifriranje. Ovdje je ključ za šifriranje javni ključ. Naime, bilo tko može šifrirati poruku pomoću njega, ali samo osoba koja ima odgovaraju ključ za dešifriranje (privatni ili tajni ključ) može dešifrirati tu poruku.

## A.2 Kriptosustavi s javnim ključem

Ideja javnog ključa se sastoji u tome da se konstruiraju kriptosustavi kod kojih bi iz poznavanja funkcije šifriranja $e_k$ bilo praktički nemoguće izračunati funkciju dešifriranja $d_k$. Tada bi funkcija $e_k$ mogla biti javna.

U provedbi ove ideje ključnu ulogu igraju tzv. osobne jednosmjerne funkcije. Za funkciju $f$ kažemo da je jednosmjerna (one-way) ako je $f$ lako, a $f^{-1}$ teško izračunati. Ako je pritom $f^{-1}$ lako izračunati ukoliko nam je poznat neki dodatni podatak, onda f nazivamo osobna jednosmjerna funkcija.

**Definicija A.2.1.** *Kriptosustav s javnim ključem se sastoji od dviju familija $\{e_K\}$ i $\{d_K\}$ funkcija za šifriranje i dešifriranje (ovdje K prolazi skupom svih mogućih korisnika) sa svojstvom:*

1. *Za svaki K je $d_K$ inverz $e_K$.*
2. *Za svaki K je $e_K$ javan, ali je $d_K$ poznat samo osobi K.*
3. *Za svaki K je $e_K$ osobna jednosmjerna funkcija.*

*$e_K$ se zove javni ključ, a $d_K$ tajni ili osobni ključ.*

Ako pošiljalac A želi poslati poruku x primaocu B, onda B najprije pošalje A svoj javni ključ $e_B$. Potom A šifrira svoju poruku pomoću $e_B$ i pošalje primaocu šifrat $y = e_B(x)$. Konačno, B dešifrira šifrat koristeći svoj tajni ključ $d_B$: $d_B(y) = d_B(e_B(x)) = x$.

Ovdje se može postaviti pitanje kako osoba B može biti sigurna da joj je upravo osoba A poslala poruku. Naime, svatko ima pristup funkciji eB, pa se može lažno predstaviti

kao osoba A. Dakle, postavlja se pitanje vjerodostojnosti ili autentičnosti poruke. Neki kriptosustavi omogućavaju čak da korisnici digitalno potpišu svoju poruku. To je važno zbog toga što tada A ne može kasnije zanijekati da je upravo on poslao konkretnu poruku. Glavne prednosti kriptosustava s javnim ključem u usporedbi sa simetričnima su:

- nema potrebe za sigurnim komunikacijskim kanalom za razmjenu ključeva
- za komunikaciju grupe od $N$ ljudi treba $2N$ ključeva, za razliku od $N(N-1)/2$ ključeva kod simetričnog kriptosustava
- mogućnost potpisa poruke

# Dodatak B

# Digitalni potpis

Pretpostavimo da dvoje ljudi A i B žele razmijenjivati potpisane poruke tj. žele biti sigurni u identitet osobe od koje su poruku dobili. Kao prvo, obje osobe kreiraju par komplementarnih ključeva, javni i tajni ključ. važno je naglasiti da se poznavanjem javnog ključa ne može izračunati tajni ključ u nekom razumnom vremenu. Nakon kreiranja ključeva osobe A i B razmjenjuju svoje javne ključeve, a potom pošiljatelj A koristi svoj tajni ključ za šifriranje sažetka poruke koji je izračunao nekom od hash funkcija. Hash funkcija je funkcija koja iz zadane poruke računa sažetak fiksne duljine, obično od 128 do 256 bita. Kada primatelj B uspije dešifrirati sažetak poruke javnim ključem pošiljatelja A on još računa i sažetak primljene poruke koji potom uspoređuje s upravo dešifriranim, i ako je izračunati sažetak jednak onom dešifriranom, primatelj može biti siguran u porijeklo poruke jer je poruka mogla biti šifrirana jedino tajnim ključem pošiljatelja A, kao i u integritet poruke.

U cijeloj proceduri samo je jedna stvar slaba karika. Moramo biti apsolutno sigurni da javni ključ za koji mislimo da pripada pošiljatelju A zaista i pripada pošiljatelju A. Naime, ukoliko primatelj B ima javni ključ pošiljatelja C, a vjeruje da ključ pripada pošiljatelju A, tada je pošiljatelj C u mogućnosti krivotvoriti podatke pošiljatelja A.

Opisani problem rješava se na način da se uvodi povjerljiva stranka, PS. Pretpostavka je da povjerljivoj stranci sve ostale stranke vjeruju, te da svoje javne ključeve osobno odnesu na potpisivanje, s time da im PS prethodno provjeri uobičajene fizičke dokumente. U tom slučaju PS koristi svoj tajni ključ (javni ključ PS svima je poznat) za potpisivanje javnog ključa te time garantira svima ostalima ispravnost potpisanog javnog ključa.

Postoji i druga mogućnost, a to je da PS svima generira par ključeva, te uz prethodnu fizičku autentifikaciju, dodjeljuje ključeve. U tom slučaju svatko tko bi htio provjeriti ispravnost potpisa osobe O morao bi u bazi javnih ključeva (koju čuva PS) pronaći javni ključ

osobe O i potom tim ključem pokušati dešifrirati primljene podatke. Nedostatak ovog drugog modela je taj što u tom slučaju PS posjeduje i tajne ključeve što predstavlja znatan sigurnosni problem ako se isti par ključeva koristi osim za potpisivanje i za šifriranje podataka.

Kao logičan izbor za PS nameću se državne ustanove, sudovi i javni bilježnici, iako su trenutno jedine takve ustanove tvrtke poput Twawte i Verisign koje izdaju certifikate potrebne tvrtkama koje žele svojim klijentima osigurati sigurnu vezu prema svom web poslužitelju SSL protokolom.

Osim modela jedne centralne povjerljive osobe postoji i neka vrsta hijerarhijskog modela , kod kojeg je svaki korisnik u mogućnosti potpisati javne ključeve drugih osoba (za koje je siguran da pripadaju pravim osobama) te time garantirati drugima, koji su sigurni u ispravnost njegovog javnog ključa, ispravnost potpisanih ključeva.

## Potpisi i zakoni

Zakonske regulative (zemalja koje imaju zakon o digitalnom potpisu) ne određuju niti jednu tehnologiju potpisivanja kao dominantnu, već samo donose propise kojih se svaka od tehnologija mora pridržavati. Od digitalnog potpisa očekuje se da bude jedinstven osobi koja ga koristi, da se može provjeriti kome pripada odnosno da li zaista pripada osobi koja ga je koristila, da je u potpunoj kontroli osobe koja ga koristi te da potvrđuje i sebe i podatke koje potpisuje.

Već iz ovog vidimo da postoji znatna prednost digitalnog potpisa nad klasičnim metodama autentifikacije. Najveća prednost je ta što se valjanost potpisa provjerava svaki put pri primitku dokumenta, za razliku od klasičnih potpisa koji se provjeravaju tek na sudu, kad se prijevara već odigrala. Osim ove prednosti postoji još jedna značajna prednost, a to je nemogućnost naknadne izmjene potpisanog dokumenta, kao i nemogućnost potpisivanja praznih dokumenata. Ipak, ukoliko krivotvoritelj uspije doći do tajnog ključa, tada bez ikakvih problema može falsificirati podatke bez da postoji i najmanja mogućnost utvrđivanja različitosti takvog potpisa od pravog potpisa, što kod klasičnih metoda ipak nije slučaj.

## B.1 Kriptografski temelji digitalnog potpisa

Današnje tehnike digitalnog potpisivanja temelje se na algoritmima asimetrične kriptografije, poznate još i pod nazivom kriptografija javnog ključa. Algoritme javnog ključa možemo podijeliti u tri osnovne grupe:

1. algoritmi temeljeni na praktičnoj nemogućnosti faktoriziranja velikih prostih brojeva (RSA)
2. algoritmi temeljeni na praktičnoj nemogućnosti izračunavanja diskretnih logaritama (Diffie-Hellman protokol, DSA)
3. algoritmi temeljeni na eliptičnim krivuljama

Osim ovih algoritama postoji još i nekolicina rjeđe korištenih, temeljenih na praktičnoj nemogućnosti utvrđivanja sadržaja ruksaka, no većina današnjih komercijalnih implementacija se može svrstati u jednu od tri glavne kategorije.

## B.2 RSA kriptosustav

Prvi, a ujedno i najpopularniji i najšire korišteni kriptosustav s javnim ključem je RSA kriptosustav koji su izumili Ron Rivest, Adi Shamir i Len Adleman 1977. godine. Njegova sigurnost je zasnovana na teškoći faktorizacije velikih prirodnih brojeva. Slijedi precizna definicija RSA kriptosustava.

**Definicija B.2.1.** *Neka je $n = pq$, gdje su $p$ i $q$ prosti brojevi. Neka je $P = C = Z_n$, te*

$$K = \{(n, p, q, d, e) : n = pq,\ p,\ q \text{ prosti},\ de \equiv 1(\mathrm{mod}\varphi(n))\}$$

*. Za $K = (n, p, q, d, e) \in K$ definiramo*

$$e_K(x) = x^e \bmod n \text{ i } d_K(y) = y^d \bmod n,\ x,\ y\ \in \mathbb{Z}_n.$$

*Vrijednosti n i e su javne, a vrijednosti p, q i d su tajne.*

Ovdje je $\varphi(n)$ Eulerova funkcija, tj. broj brojeva u nizu 1, 2, ... , n koji su relativno prosti s n.

Uvjerimo se da su funkcije $e_K$ i $d_K$ jedna drugoj inverzne.

Imamo: $d_k(e_k(x)) \equiv x^{de} \bmod n$. Iz $de \equiv 1 \bmod \varphi(n)$ slijedi da postoji prirodan broj $t$ takav da je $de = t\varphi(n) + 1$. Sada je

$$x^{de} = x^{t\varphi(n)+1} = \left[x^{\varphi(n)}\right]^t \cdot x \equiv x \bmod n$$

ako je $(n, x) = 1$ (prema Eulerovom teoremu).

Ako je $(n, x) = n$, onda je $x^{de} \equiv x \equiv 0 \bmod n$; ako je $(n, x) = p$, onda je $x^{de} \equiv x \equiv 0(\bmod p)$ i $x^{de} = \left[x^{q-1}\right]^{(p-1)t} \cdot x \equiv x(\bmod q)$, pa je $x^{de} \equiv x(\bmod n)$. Slučaj $(n, x) = q$ je potpuno analogan.

Prema tome, zaista je $x^{de} \equiv x(\bmod n)$, što znači da je $d_k(e_K(x)) = x$

**Primjer B.2.2.** *Uzmimo $p = 3$ i $q = 11$. Tada je $n = 33$ i $\varphi(n) = 20$. Eksponent e mora biti relativno prost s 20, pa recimo da je $e = 7$. Tada je $d = 3$. Sada je $(n, e) = (33, 7)$ naš javni ključ. Pretpostavimo da nam netko želi poslati poruku $x = 17$. To znači da treba izračunati $e_K(x) = 17^7$ mod 33:*

$$17^7 = 17 \cdot 17^2 \cdot 17^4 \equiv 17 \cdot 25 \cdot (-2) \equiv -25 \equiv 8(\textit{ mod } 33)$$

*Dakle, šifrat je* $y = e_K(x) = 8$. *Kada primimo ovaj šifrat, dešifriramo ga pomoću tajnog ključa d:*

$$x = d_K(y) = 8^3 = 8 \cdot 8^2 \equiv 8 \cdot (-2) \equiv 17(\ mod\ 33)$$

*Dakle,* $x = 17$.

Sigurnost RSA leži u pretpostavci da je funkcija $e_K(x) = x^e \bmod n$ jednosmjerna. Dodatni podatak koji omogućava dešifriranje je poznavanje faktorizacije $n = pq$, jer je tada lako izračunati $\varphi(n) = (p-1)(q-1)$, te dobiti eksponent $d$ iz $de \equiv 1(\bmod \varphi(n))$, pomoću Euklidovog algoritma.

Opišimo sada malo detaljnije postupak kojim korisnik izabire parametre u RSA kriptosustavu.

1. Izabiremo tajno dva velika prosta broja $p$ i $q$ od oko 100 znamenaka, tako da $q$ ima nekoliko znamenaka više od $p$. To radimo tako da pomoću nekog generatora slučajnih brojeva generiramo prirodan broj $m$ s traženim brojem znamenaka, a zatim korištenjem nekog testa za testiranje prostosti (opisat ćemo ih u sljedećem poglavlju) tražimo prvi prosti broj veći ili jednak $m$. Po teoremu o distribuciji prostih brojeva, možemo očekivati da ćemo trebati testirati $O(\log m)$ brojeva dok ne nađemo prvi prosti broj.

2. Izračunamo $n = pq$ i $\varphi(n) = (p-1)(q-1) = n+1-p-q$. (Za to nam treba $O(\log^2 n)$ operacija.)

3. Izaberemo na slučajan način broj $e$ takav da je $e < \varphi(n)$ i $(\varphi(n), e) = 1$. To se može napraviti slično kao pod 1. Nakon toga tajno izračunamo $d$ tako da je $de \equiv 1(\bmod \varphi(n))$, tj. $d \equiv e^{-1}(\bmod \varphi(n))$. To se radi pomoću Euklidovog algoritma i za to nam treba $O(\log^3 n)$ operacija.

4. Stavimo ključ za šifriranje $(n, e)$ u javni direktorij.

Za efikasnost RSA kriptosustava, važna je činjenica da se modularno potenciranje može izvesti vrlo efikasno. Pokažimo kako se efikasno računa $e_K(x) = x^e \bmod n$. To se radi tzv. metodom "kvadriraj i množi". Najprije $e$ prikažemo u bazi 2:

$$e = e_0 + 2 \cdot e_1 + ... + 2^{s-1} \cdot e_{S-1}$$

a potom primjenimo sljedeći algoritam:

$y = 1$
za $i = s - 1, ..., 1, 0$
. $y = y^2 \bmod n$
. ako je $e_i = 1$ onda je $y = y \cdot x \bmod n$

Očito je ukupan broj množenja $\leq 2s$, pa je ukupan broj operacija $O(\log e \cdot \log^2 n)$. To znači da je ovaj algoritam polinoman.

Jedan očit napad na RSA je faktorizacija od $n$. Ako napadač faktorizira $n$, onda može naći $\varphi(n)$ i $d$. Trenutno najbrži algoritmi za faktorizaciju trebaju

$$exp(O((\log n)^{\frac{1}{3}}(\log \log n)^{\frac{2}{3}})$$

operacija, tako da su brojevi od preko 200 znamenaka za sada sigurni od ovog napada. Dakle, nije poznat niti jedan polinomijalni algoritam za faktorizaciju. Ovdje ipak treba reći da je u nekim slučajevima $n$ puno lakše faktorizirati, pa takve $n$-ove treba izbjegavati. Takav je slučaj npr. ako su $p$ i $q$ jako blizu jedan drugoga ili ako $p - 1$ i $q - 1$ imaju samo male proste faktore.

Važno je napomenuti da ako napadač otkrije tajni eksponent $d$, onda nije dovoljno promijeniti samo eksponent $e$, već moramo promijeniti i $n$.

## B.3 SHA-256

Kao što je već ranije napisano digitalni potpis mora garantirati vjerodostojnost potpisane informacije. Upravo se za to koriste hash funkcije koje izračunavaju sažetak poruke. Jedna od hash funkcija je i SHA-256 koja se koristi u Bitcoin protokolu i koja će ovdje biti detaljno opisana.

Poruku čiju hash vrijednost želimo izračunat prvo proširimo tako da je konačna dužina u bitovima djeljiva sa 512, a zatim ju podijelimo u blokove duzine 512 bitova $M^{(1)}$, $M^{(2)}$, ..., $M^{(N)}$. Blokovi poruke se obrađuju jedan po jedan: Počevši sa predodređenom početnom hash vrijednošću $H^{(0)}$, slijedno se računa

$$H^{(i)} = H^{(i)} + C_M^{(i)}(H^{(i-1)})$$

gdje je C SHA-256 kompresijska funkcija, a '+' je word-wise mod $2^{32}$ zbrajanje. $H^{(N)}$ je hash vrijednost poruke $M$.

SHA-256 kompresijska funkcija radi sa 512-bitnim blokovima poruke i 256-bitnim međurezultatima hash vrijednosti. SHA-256 je u biti 256-bitni kriptografski algoritam koji kriptira međurezultate hash vrijednosti koristeći blok poruke kao ključ.
Početna hash vrijednost $H^{(0)}$ je slijedeći niz 32-bitnih riječi:

$$H_1^{(0)} = 6a09e667$$
$$H_2^{(0)} = bb67ae85$$
$$H_3^{(0)} = a54ff53a$$
$$H_4^{(0)} = 510e527f$$
$$H_5^{(0)} = 9b05688c$$
$$H_6^{(0)} = 1f83d9ab$$
$$H_7^{(0)} = 5be0cd19$$

Izračunavanje hash vrijednosti počinje proširivanjem poruke. Pretpostavimo da je duljina poruke $M$, u bitovima $l$. Dodajemo bit '1' na kraj poruke, i zatim $k$ nula bitova, gdje je $k$ najmanje nenegativno rješenje jednadžbe $l+1+k = 448 \mod 512$. Na to se dodaje 64-bitni blok koji je jednak broju $l$ zapisanom binarno. Na primjer, poruka "abc" (u 8-bitnom ASCII kodu) ima dužinu $8 * 32 = 24$ pa se proširuje prvo sa '1' i zatim sa 448–(24 + 1) = 443 nula bitova, nakon čega dobijemo proširenu poruku od 512-bitova :

$$01100001\ 01100010\ 1\ \underbrace{00...0}_{423}\ \underbrace{00...011000}_{64}$$

Zatim podijelimo poruku na $N$ 512-bitnih blokova $M^{(1)}, M^{(2)}, ..., M^{(N)}$. Prvih 32 bita i-tog bloka su označena sa $M_0^{(i)}$, sljedećih 32 bita su $M_1^{(i)}$, i tako do $M_{15}^{(i)}$.
Nastavak računanja prikazan je sljedećim pseudokodom:

Za $i = 1$ do $N$ ( $N$ je broj blokova proširene poruke )
{

- Inicijalizacija registara $a, b, c, d, e, f, g, h$ sa $(i-1)$-im međurezultatom hash vrijednosti.

$$a \leftarrow H_1^{(i-1)}$$
$$b \leftarrow H_2^{(i-1)}$$
$$.$$
$$.$$
$$.$$
$$h \leftarrow H_8^{(i-1)}$$

- Za $j = 0 do 63$ Izračunaj $Ch(e, f, g)$, $Maj(a, b, c)$, $\Sigma_0(a)$, $\Sigma_1(e)$, i $W_j$ (definicije su niže u tekstu).

$$T_1 \leftarrow h + \Sigma_1(e) + Ch(e, f, g) + K_j + W_j$$
$$T_2 \leftarrow \Sigma_0(a) + Maj(a, b, c)$$
$$h \leftarrow g$$
$$g \leftarrow f$$
$$f \leftarrow e$$
$$e \leftarrow d + T_1$$
$$d \leftarrow c$$
$$c \leftarrow b$$
$$b \leftarrow a$$
$$a \leftarrow T_1 + T_2$$

- Računanje *i*-tog međurezultata hash vrijednosti

$$H_1^{(i)} \leftarrow a + H_1^{(i-1)}$$
$$H_2^{(i)} \leftarrow b + H_2^{(i-1)}$$
$$.$$
$$.$$
$$.$$
$$H_8^{(i)} \leftarrow h + H_8^{(i-1)}$$

}
$H^{(N)} = (H_1^{(N)}, H_2^{(N)}, ..., H_8^{(N)})$ je hash vrijednost poruke M.

U SHA-256 koristi se 6 logičkih funkcija. Sve funkcije rade sa 32-bitnim riječima i vračaju 32-bitni rezultat.

$$Ch(x, y, z) = (x \text{ AND } y) \text{ XOR } (\neg x \text{ AND } z)$$
$$May(x, y, z) = (x \text{ AND } y) \text{ XOR } (x \text{ AND } z) \text{ XOR } (y \text{ AND } z)$$
$$\sum_{(0)}^{(256)}(x) = ROTR^2(x) \text{ XOR } ROTR^{13}(x) \text{ XOR } ROTR^{22}(x)$$
$$\sum_{(1)}^{(256)}(x) = ROTR^6(x) \text{ XOR } ROTR^{11}(x) \text{ XOR } ROTR^{25}(x)$$
$$\sigma_{(0)}^{(256)}(x) = ROTR^7(x) \text{ XOR } ROTR^{18}(x) \text{ XOR } SHR^3(x)$$
$$\sigma_{(1)}^{(256)}(x) = ROTR^{17}(x) \text{ XOR } ROTR^{19}(x) \text{ XOR } SHR^{10}(x)$$

$ROTR^{(n)}(x)$ predstavlja rotaciju u desno za n bitova, a $SHR^{(n)}(x)$ posmak u desno za n bitova.

Prošireni blokovi $W_0, W_1, ..., W_{63}$ se računaju prema sljedećem rasporedu:

$$W_j = M_j^{(i)} \text{ , za } j = 0, ..., 15$$
$$W_j = \sigma_1(W_{j-2}) + W_{j-7} + \sigma_0(W_{j-15}) + W_{j-16}$$

.

# Bibliografija

[1] Andreas M. Antonopoulos, *Mastering Bitcoin*, O'Reilly Media Inc., 2015.

[2] Satoshi Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*, (2008), `https://bitcoin.org/bitcoin.pdf`.

[3] M. Nielsen, *How the Bitcoin protocol actually works*, (2013), `http://www.michaelnielsen.org/ddi/how-the-bitcoin-protocol-actually-works/`.

# Sažetak

Bitcoin je prvi primjer kriptovalute. Bitcoin mreža je započela s radom 2009. godine i danas broji oko osam milijuna korisnika. Bitcoin je decentralizirana valuta te kao takva nije pod kontrolom nikakve središnje banke. Transakcije u Bitcoin mreži potvrđuju se tako da korisnici dođu na konsenzus o valjanosti određene transakcije. Bitcoinovi se stvaraju kroz proces rudarenja koji se sastoji od traženja rješenja složenog matematičkog problema.

# Summary

Bitcoin is the first example of cryptocurrency. Bitcoin network begun working in 2009. and today consists of approximately eight million users. Bitcoin is a decentralised currency and as such it is not under control of any central authority. Transactions in Bitcoin network are verified by users who reach a consenzus about validation of certain transaction. New Bicoins are created through a process called mining which includes finding a solution of a difficult mathematical problem.

# Životopis

Rođena sam u Rijeci 12.11.1989. Školske godine 1996/1997. upisala sam se u prvi razred Osnovne skole "Gornja Vežica" u Rijeci. Školske godine 2003/2004. upisala sam Prirodoslovno-matematičku gimnaziju "Andrija Mohorovičić" u Rijeci. Sudjelovanjem na državnim natjecanjima u drugom i trećem razredu izborila sam izravan upis na studij matematike na Prirodoslovno-Matematičkom fakultetu u Zagrebu. Akademske godine 2008/2009. upisala sam se na preddiplomski sveučilišni studij matematike na Prirodoslovno-matematičkom fakultetu u Zagrebu kojeg završavam 31.07.2012. čime sam stekla akademski naziv Baccalaurea matematike (sveučilišna prvostupnica matematike). Akademske godine 2012/13. upisala sam diplomski studij Matematika i računarstvo na Prirodoslovno-matematičkom fakultetu u Zagrebu.